

SYSTEM AND METHOD FOR PROXIMITY SEARCHING POSITION INFORMATION USING A PROXIMITY PARAMETER

Inventor: Jeremy S. Cooper

BACKGROUND OF THE INVENTION

[0001] This application claims the benefit of U.S. Provisional Application No. 60/199,551, filed April 25, 2000.

Field of the Invention

[0002] The present invention relates to computer networking and communication, including Web-based communications and commerce.

Related Art

[0003] A proximity search has utility in many computer and business related applications. A proximity search refers to a search that identifies any candidate geographical positions that are near a geographical position of interest. In one known application, each candidate geographical position can be represented by position field information associated with a database record. A conventional proximity search includes the computer implemented steps of (a) calculating a great circle separation distance between the position of interest and each database position, and then (b) comparing each separation distance to a proximity parameter or search range to determine which of the candidate position are near, that is, proximate, the position of interest. The great circle separation distance is the distance between two points on the surface of the Earth along a great circle or circumference of the Earth, and therefore represents the shortest distance between the two points on the surface of the Earth.

[0004] Because great circle separation distances are used, the above mentioned proximity search accurately identifies candidate positions within the search range

of the position of interest. However, calculating the great circle separation distances is computationally intensive because of the relatively complex geometric calculations involved. Thus, proximity search accuracy is achieved at the expensive of a heavy computational burden associated with calculating great circle separation distances. In an application including hundreds of thousands, or even millions, of database positions, calculating a correspondingly large number of great circle separation distances disadvantageously imposes an onerous computational burden, and thus consumes valuable computer processing time.

[0005] Different applications require proximity searches having different characteristics. For example, an application that requires fast responses to user queries, correspondingly requires rapid proximity searches. Such an application may relax proximity search accuracy so as to increase computational efficiency and speed. One such application requires a time efficient proximity search of millions of candidate positions, as mentioned above. In another application, the search range or proximity parameter is in a format that is incompatible with a format of the position field information in the database, thus complicating the processing of a query requiring a proximity search.

[0006] Therefore, there is a need for a time efficient proximity search that identifies candidate geographical positions that are near a geographical position of interest. There is a related need for a proximity search that searches such candidate geographical position when represented by position field information associated with a database record.

[0007] There is also a need for a proximity search that identifies the candidate geographical positions within a search range, specified as a proximity parameter, of the position of interest. There is a related need for a proximity search capable of searching for the candidate positions when the proximity parameter and the position field information have incompatible formats.

[0008] There is a further need for a proximity search that is responsive to user queries.

SUMMARY OF THE INVENTION

[0009] The present invention provides a computer implemented proximity searcher that searches position field information (representing geographical positions) stored in database records, to identify such positions falling within a predetermined search range of a position of interest. The proximity searcher of the present invention reduces the computational burden associated with conventional proximity search routines, such as the great circle calculation mentioned above, so as to perform time efficient proximity searches to identify candidate geographical positions that are near the position of interest. The proximity searcher identifies the candidate geographical positions within a search range, specified as a proximity parameter, of the position of interest. The proximity parameter and the position field information can have incompatible formats. In one embodiment of the present invention, the proximity searcher is responsive to user queries.

[0010] The present invention provides a method of performing a proximity search, wherein the method includes the step of receiving a proximity parameter defining a search area around a predetermined position (the position of interest). A set of latitudes and longitudes approximating the search area are calculated based on the proximity parameter. The set of latitudes and longitudes are compared to position field information in a plurality of records stored in a database. The method determines which of the plurality of records include position information within the search area based on the comparison step.

[0011] In one embodiment, the proximity parameter is a search radius defining a circular search area centered around the predetermined position. The set of latitudes and longitudes are calculated to define a smallest square search area into which the circular search area can fit, and that approximates the proximity parameter defined circular search area. The method includes comparing a latitude and a longitude associated with each of the plurality of records to a latitude range and a longitude ranges covered by the smallest square search area to determine

which of the plurality of records include position information within the square search area.

[0012] The present invention further provides a system and a computer program product for performing proximity searches in accordance with the above mentioned method of performing same.

[0013] Additional features and advantages of the present invention, as well as the structure and operation of various embodiments of the present invention, are described in detail below with reference to the accompanying drawings.

BRIEF DESCRIPTION OF THE FIGURES

[0014] The accompanying drawings, which are incorporated herein and form part of the specification, illustrate the present invention and, together with the description, further serve to explain the principles of the invention and to enable a person skilled in the pertinent art make and use the invention.

[0015] FIG. 1 is an illustration of an exemplary operating environment of the present invention.

[0016] FIG. 2 is an illustration of a high level method performed by a proximity searcher, according to an embodiment of the present invention.

[0017] FIG. 3 is an illustration of an exemplary record table stored in a database of FIG. 1, and accessible to the proximity searcher, according to the present invention.

[0018] FIG. 4 is a diagrammatic illustration of a search request overlaid on a map-outline of the United States, according to the present invention.

[0019] FIG. 5 is a cross sectional view of the Earth, wherein a search location of FIG. 4 is depicted at a latitude in the Northern Hemisphere.

[0020] FIG. 6 is a perspective view of the Northern Hemisphere depicted in FIG. 5.

- [0021] FIG. 7 is an illustration of an exemplary series of detailed method steps for calculating a set of latitude and longitudes corresponding to a search area approximating a proximity parameter defined search area.
- [0022] FIG. 8A is a diagram of an example internetwork environment according to the present invention.
- [0023] FIG. 8B is an illustration of a simplified four-layered communication model supporting Web commerce including an application layer, a transport layer, an Internet layer, and a physical layer.
- [0024] FIG. 8C is an exemplary computer architecture on which the present invention can be implemented.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

- [0025] FIG. 1 is an illustration of an exemplary operating environment 100 of the present invention. Operating environment 100 is also described in further detail below in connection with FIGs. 8A, 8B and 8C. In one embodiment, environment 100 is a client-server environment, and includes a plurality of user computers (also referred to as "clients") 102₁, 102₂ and 102_n coupled to a communication network 104. A client application, such as a browser, executes on each of the user computers 102. Communication network 104 can be any known communication network, including the Internet, local area networks (LANs), the Public Switch Telephone Network (PSTN), and so on.
- [0026] A server computer (also referred to as a "server") 106 associated with a service provider 107 is also coupled to communication network 104. A server application executes on server 106, and includes a proximity searcher 108 for performing proximity searches, according to the present invention. Server 106 is coupled to a database 110 for storing position field information in records that can be proximity searched by proximity searcher 108. A method of proximity

searching associated with proximity searcher 108 and database 110 is described in further detail below.

[0027] In a typical client-server application of the present invention, clients 102 can send information requests to server 106 via communication network 104. In response to an information request received from one of client computers 102, server 106 formulates an information response and sends the information response back to the requesting client computer via communication network 104.

[0028] FIG. 2 is an illustration of a high level method 200 performed by proximity searcher 108, according to an embodiment of the present invention. Method 200 is initiated when one of client computers 102 sends an information request to server 106 requiring a proximity search. At a first step 205, the information request is received by server 106. The information request is associated with a proximity parameter defining a search range or area around (that is, encompassing) a position of interest (also referred to as a "predetermined position"). In one embodiment, the proximity parameter represents a search radius defining a circular search area centered around the predetermined position. The search radius can be in any length based units, such as meters, feet, miles, etc.

[0029] In an alternative embodiment, method 200 is initiated when a computer process executing on server 106, for example, sends a search request to proximity searcher 108 requesting a proximity search. The search request includes the proximity parameter and is associated with a position of interest, as mentioned above.

[0030] At a next step 210, a set of latitudes and longitudes approximating the circular search area are calculated. In one embodiment, latitudes and longitudes are calculated to define a smallest square search area into which the circular search area can fit. The square search area has a width and a height equal to at least twice the search radius (that is, equal to a search diameter of the circular search area).

[0031] At a next step 215, the calculated set of latitudes and longitudes are compared to position field information, specifying locations on the surface of the

Earth, in a plurality of records stored in database 110. In one embodiment, the position field information comprises latitude and longitude information, and step 215 includes comparing the latitude and longitude ranges covered by the square search area to the respective latitudes and longitudes associated with each of the records in database 110. Mapping the circular search area to a set of latitudes and longitudes using the proximity parameter prior to comparison step 215 advantageously leads to a relatively straight forward and time efficient comparison of latitude and longitude information at step 215. In other words, mapping step 210 is a technique for converting the proximity parameter (and thus, the circular area defined thereby) into a format compatible with database position field information against which the proximity parameter must be compared. Step 210 is a "one-time" mapping step that advantageously avoids converting each of the latitudes and longitudes - in perhaps millions of database records - to some other format.

[0032] At a next step 220, it is determined which of the database records include position field information within the circular search area based on comparison step 215. In one embodiment, position field information (database positions) falling within the square search area are determined to be within the circular search area. It is to be understood that other determining criteria could be used to determine whether a database position falls within the circular search area. For example, database positions falling within a predetermined distance (that is, within a predetermined range of latitudes and longitudes) of the square search area could be determined to be within the circular search area.

[0033] At a next step 225, a search result is sent to the requesting client to fulfill the information request received at step 205. If method 200 was initiated by a search request from a process executing on server 106, proximity searcher 108 returns the search result to the requesting process. In either case, the search result is based on the database records determined to have position information within the circular search area at step 220.

[0034] The above described method is now described in further detail in the context of an example scenario, wherein service provider 107 provides a service for matching an employer having job opportunities (also known as "job postings") with a list of prospective employee candidates who reside near or "proximate" the employer. In the example scenario, database 110 is populated with employee candidate resume information. FIG. 3 is an illustration of an exemplary record table 300 stored in database 110. The information stored in table 300 includes information taken from candidate resumes. Table 300 includes a city column 302 for listing city or town identifiers representing candidate residences. A position column 304 is provided for listing the positions or locations of the candidate cities listed in column 302. Position column 304 is further subdivided into a latitude column 306 and a longitude column 308 for respectively listing the candidate city positions in terms of latitude and longitude. Table 300 optionally includes a city zip code column 310 and an "other" column 312 for listing any other desirable information associated with employee candidates.

[0035] Table 300 includes a plurality of rows 350_1 , 350_2 and 350_n . Each of the rows 350 corresponds to a record for storing candidate information in database 110. For example, row 350_1 includes a field 352 for storing a city identifier "aaa", fields 354 and 356 (referred to as "position fields") for respectively storing a latitude A_{LAT} and a longitude A_{LON} (collectively referred to as "position field information") of city aaa, and a field 360 for storing a zip code "21100" of city aaa. In one embodiment, the latitudes and longitudes are represented in radians.

[0036] In the example scenario, an employer "XYZ" has job openings at an employer location, for example, in a city within the United States. Using one of client computers 102, employer XYZ submits an information request (in this case, a search request) to server 106 to identify all of the employee candidates residing within a predetermined search radius R_s of the employer location. In an alternative scenario, an employee candidate, instead of the employer, submits an information request to identify any job postings proximate the candidate. An example search radius R_s associated with the search request may be 50 or 75

miles. In one embodiment the employer can specify search radius R_s (that is, the proximity parameter) in the information request. Alternatively, one or more default proximity parameters can be provided by server 106 in response to the information request.

[0037] In the information request, the employer location can be expressed as an entire address with a zip code, or just the zip code where employer XYZ is located. FIG. 4 is an illustration of a map-outline of the United States 400, wherein the information request is diagrammatically illustrated. Using a commercially available utility program at server 106, the zip code of company XYZ in the information request is easily mapped to a representative position C_1 expressed in terms of, for example, a latitude C_{1LAT} and a longitude C_{1LON} . Via the information request, employer XYZ wishes to identify any employee candidates residing within a circular search area 405 having a radius R_s and centered around location C_1 (C_{1LAT} , C_{1LON}). Candidates may reside in cities or towns located, for example, at positions 412 within circular search area 405.

[0038] Still with reference to FIG. 4, after receiving the above mentioned information request, proximity searcher 108 calculates a set of latitudes and longitudes corresponding to four corner points P_1 (P_{1LAT} , P_{1LON}), P_2 (P_{2LAT} , P_{2LON}), P_3 (P_{3LAT} , P_{3LON}), and P_4 (P_{4LAT} , P_{4LON}) defining a smallest square search area 415 into which circular search area 405 can fit. In other words, proximity searcher 108 maps circular search area 405 to square search area 415 defined by corner points P_1 - P_4 in terms of latitude and longitude. As depicted in FIG. 4, corner points P_1 , P_2 , P_3 and P_4 respectively correspond to a North-East corner, a South-East corner, a South-West corner and a North-West corner of square search area 415. Square search area 415 has a width W and a height H , each equal to a diameter D_s (referred to as search diameter D_s) of circular search area 405.

[0039] The step of calculating the set of latitudes and longitudes corresponding to corner points P_1 through P_4 is now described in further detail with reference to FIGs. 5-8. FIG. 5 is a cross sectional view of the Earth 500, wherein an axis line 502 lying in an equatorial plane and a North-South axis line 503 are depicted.

Location C_1 is depicted at a latitude C_{ILAT} in the Northern Hemisphere. R_E represents the radius of the Earth, and R_L represents the radius of a latitude ring (not shown) at latitude C_{ILAT} and coinciding with location C_1 . Location C_1 also lies on a longitude ring 504 defining a great circle about the surface of the earth. Since circular and square search areas 405 and 415 are small compared to the curvature of the Earth's surface, search areas 405 and 415 approximate planar areas. Thus, for purposes of the present invention, search areas 405, 415 can be considered as either planar or non-planar areas.

[0040] FIG. 6 is a perspective view of the Northern Hemisphere depicted in FIG. 5. The elements depicted in FIG. 6 are not drawn to scale. Circular and square search areas 405 and 415 are centered around location C_1 in the Northern Hemisphere. A latitude ring 602 at latitude C_{ILAT} has a circumference C_L passing through both the circular and square search areas 405, 415. The curvature of latitude ring 602 is highly exaggerated in FIG. 6. Circumference C_L of latitude ring 602 is represented by the following equations:

$$C_L = 2\pi R_L, \text{ where } R_L = R_E \cdot \cos(C_{ILAT}) \text{ (from FIG. 5), and}$$

therefore

$$C_L = 2\pi(R_E \cdot \cos(C_{ILAT}))$$

[0041] A segment 604 of latitude ring 602, having a length equal to search diameter D_s (and the width W of square search area 415), bisects the height of square search area 415. Segment 604 subtends an angular width Δ_{LON} at axis line 503. Δ_{LON} represents a longitudinal angular measure or extent of both circular and square search areas 405 and 415. Δ_{LON} is represented by the following equation:

$$\Delta_{LON} = 2\pi(D_s / C_L) \text{ radians}$$

[0042] A segment 606 of longitude ring 504, also having a length equal to D_s (and the height W of square 415), bisects the width of square search area 415.

Segment 606 subtends an angular height Δ_{LAT} at an intersection between axes 502 and 503. Δ_{LAT} represents an latitudinal angular measure of both circular and square search areas 405 and 415. Δ_{LAT} is represented by the following equation:

$$\Delta_{LAT} = 2\pi(D_S / C_E) \text{ radians}$$

[0043] The latitude and longitude coordinates defining the four corners P_1 through P_4 of square search area 415 are calculated based on C_1 (C_{ILAT} , C_{ILON}), Δ_{LAT} , and Δ_{LON} , according to the equations below.

For P_1 : $P_{1LAT} = C_{ILAT} + (\Delta_{LAT} / 2)$, and
 $P_{1LON} = C_{ILON} - (\Delta_{LON} / 2)$

For P_2 : $P_{2LAT} = C_{ILAT} - (\Delta_{LAT} / 2)$, and
 $P_{2LON} = C_{ILON} - (\Delta_{LON} / 2)$

For P_3 : $P_{3LAT} = C_{ILAT} - (\Delta_{LAT} / 2)$, and
 $P_{3LON} = C_{ILON} + (\Delta_{LON} / 2)$

For P_4 : $P_{4LAT} = C_{ILAT} + (\Delta_{LAT} / 2)$, and
 $P_{4LON} = C_{ILON} + (\Delta_{LON} / 2)$

Note that these calculations apply to the case where all of the search area 415 is located in the Northern and Western hemispheres, e.g., North America. In cases where some or all of the search area 415 is located in another hemisphere, the calculations must be modified in a manner that is known to persons of ordinary skill in the art.

[0044] FIG. 7 is an illustration of an exemplary series of detailed method steps 700 summarizing the above described method of calculating the set of latitude and longitudes corresponding to positions P_1 through P_4 defining square search area

415. Note that the series of method steps 700 expands on step 210 described above in connection with FIG. 3. With reference to FIG. 7, a first step 705 includes calculating an angular height (for example, Δ_{LAT}) of a proximity parameter defined search area (for example, circular area 405). The angular height is subtended by at least a search diameter (for example, D_s), and corresponds to a height of a search area (for example, square search area 415) approximating the parameter defined search area (also referred to as an "approximate search area").

[0045] At a next step 710, an angular width (for example, Δ_{LON}) of the parameter defined search area is calculated. The angular width is subtended by the search diameter, and corresponds to a width of the approximate search area. At a next step 715, latitudes associated with a set of corner positions defining the approximate search area are calculated based on the angular height of the approximate search area and a latitude of a predetermined center position (for example, C_1) about which the parameter defined search area is centered. At a next step 720, longitudes associated with the set of corner positions are calculated based on the angular width of the approximate search area and a longitude of the center position.

[0046] Referring again to step 220, described previously in connection with FIG. 2, once the circular search area has been mapped to the approximate search area (for example, square search area 415) defined in terms of longitude and latitudes, a straightforward comparison can be made between (a) the latitudes and longitudes defining the approximate search area, and (b) the latitude and longitude position field information (that is, database positions) in database records 350, to determine whether the database positions fall within the approximate search area. For example, with reference to the example scenario, a database position falls within the approximate search area 415 when the following two conditions are met:

(a) the database position latitude is between approximate square search area latitudes P_{1LAT} (or P_{4LAT}) and P_{2LAT} (or P_{3LAT}); and at the same time,

(b) the database position longitude is between approximate square search area longitudes P_{1LON} (or P_{2LON}) and P_{3LON} (or P_{4LON}).

[0047] Only three of the four positions P_1 through P_4 are necessary to test the above two conditions. Thus, to save computing time, only three of the four positions are calculated in one embodiment (for example, P_1 (P_{1LAT} , P_{1LON}), P_2 (P_{2LAT} , P_{2LON}) and P_3 (P_{3LAT} , P_{3LON})).

[0048] In the above described embodiment, at least three corner positions define the extent of square search area 415. However, other positions on a perimeter of the square search area can be calculated to define the extent of the square search area. For example, the square search area can be defined by a first position and a second position respectively bisecting the left (West) side and right (East) sides of the square search area, together with a third position and a fourth position respectively bisecting the top (North) and bottom (South) sides of the square search area. In other words, the first, second, third, and fourth positions define a cross centered at C_1 . In this configuration, all four positions are necessary to define the range of latitudes and longitudes covered by the square search area. Other position combinations that would be apparent to one skilled in the art are possible.

[0049] In another embodiment, the circular search area can be mapped to a non-square shaped, approximate search area defined by a set of latitudes and longitudes. For example, the circular search area can be mapped to a rectangularly shaped search area having different height and a width dimensions. Alternatively, the circular search area may be mapped to a parallelogram, a rhombus, or any other conveniently shaped search area, so long as the shape of the approximate search area facilitates a straight forward comparison between latitudes and longitudes, as described above.

Example Network Environment

[0050] The present invention can be implemented in any communication network, such as, the Internet, which supports interactive services and applications. In particular, the present invention can be implemented in any Web service, preferably a Web service supporting secure transactions, such as, the Secure Socket Layer (SSL) protocol and/or using a Secure HyperText Transport Protocol (S-HTTP). In one example, the present invention is implemented in a multi-platform (platform independent) programming language such as Java. Java-enabled browsers are used, such as, Netscape, HotJava, and Microsoft Explorer browsers. Active content Web pages can be used. Such active content Web pages can include Java applets or ActiveX controls, or any other active content technology developed now or in the future. The present invention, however, is not intended to be limited to Java or Java-enabled browsers, and can be implemented in any programming language and browser, developed now or in the future, as would be apparent to a person skilled in the art given this description. Further, the present invention is not intended to be limited to a Web-based implementation or environment and can be implemented in any communication network now or in the future, as would be apparent to a person skilled in the art given this description. Even further, the present invention can operate in the absence of a network, for example, on a computer not connected with a network.

[0051] FIG. 8A is a diagram of an example internetwork environment according to the present invention. FIG. 8A shows a communication network or combination of networks (Internet) 800 (corresponding to communication network 104 of FIG. 1) which can support the invention. Internet 800 consists of interconnected computers which supports communication between many different types of users including businesses, universities, individuals, government, and financial institutions. Internet 800 supports many different types of communication links implemented in a variety of architectures. For example,

voice and data links can be used including phone, paging, cellular, and cable TV (CATV) links. Terminal equipment can include local area networks, personal computers with modems, content servers of multi-media, audio, video, and other information, pocket organizers, Personal Data Assistants (PDAs), and set-top boxes.

[0052] Communication over a communication network such as, Internet 800, is carried out through different layers of communication. FIG. 8B shows a simplified four-layered communication model supporting Web commerce including an application layer 808, transport layer 810, Internet layer 820, physical layer 830. As would be apparent to a person skilled in the art, in practice, a number of different layers can be used depending upon a particular network design and communication application. Application layer 808 represents the different tools and information services which are used to access the information over the Internet. Such tools include, but are not limited to, telenet log-in service 801, IRC chat 802, Web service 803, and SMTP (Simple Mail Transfer Protocol) electronic mail service 806. Web service 803 allows access to HTTP documents 804, and FTP and Gopher files 805. A Secure Socket Layer (SSL) is an optional protocol used to encrypt communications between a Web browser and Web server.

[0053] Description of the example environment in these terms is provided for convenience only. It is not intended that the invention be limited to application in this example environment. In fact, after reading the following description, it will become apparent to a person skilled in the relevant art how to implement the invention in alternative environments.

Example Computer System

[0054] An example of a computer system 840 is shown in FIG. 8C. The computer system 840 represents any single or multi-processor computer. Single-threaded and multi-threaded computers can be used. Unified or distributed memory systems can be used.

computer system 840 and external devices via communications path 866. Examples of communications interface 864 can include a modem, a network interface (such as Ethernet card), a communications port, etc. Software and data transferred via communications interface 864 are in the form of signals which can be electronic, electromagnetic, optical or other signals capable of being received by communications interface 864, via communications path 866. Note that communications interface 864 provides a means by which computer system 840 can interface to a network such as the Internet.

[0059] The present invention can be implemented using software running (that is, executing) in an environment similar to that described above with respect to FIG. 8A. In this document, the term "computer program product" is used to generally refer to removable storage unit 858, a hard disk installed in hard disk drive 852, or a carrier wave carrying software over a communication path 866 (wireless link or cable) to communication interface 864. A computer useable medium can include magnetic media, optical media, or other recordable media, or media that transmits a carrier wave or other signal. These computer program products are means for providing software to computer system 840.

[0060] Computer programs (also called computer control logic) are stored in main memory 848 and/or secondary memory 850. Computer programs can also be received via communications interface 854. Such computer programs, when executed, enable the computer system 840 to perform the features of the present invention as discussed herein. In particular, the computer programs, when executed, enable the processor 844 to perform the features of the present invention, as related to proximity searching. Accordingly, such computer programs represent controllers of the computer system 840.

[0061] The present invention can be implemented as control logic in software, firmware, hardware or any combination thereof. In an embodiment where the invention is implemented using software, the software may be stored in a computer program product and loaded into computer system 840 using removable storage drive 854, hard drive 850, or interface 860. Alternatively, the computer

program product may be downloaded to computer system 840 over communications path 866. The control logic (software), when executed by the one or more processors 844, causes the processor(s) 844 to perform the functions of the invention as described herein.

[0062] In another embodiment, the invention is implemented primarily in firmware and/or hardware using, for example, hardware components such as application specific integrated circuits (ASICs). Implementation of a hardware state machine so as to perform the functions described herein will be apparent to persons skilled in the relevant art(s).

Conclusion

[0063] While various embodiments of the present invention have been described above, it should be understood that they have been presented by way of example only, and not limitation. It will be understood by those skilled in the art that various changes in form and details may be made therein without departing from the spirit and scope of the invention as defined in the appended claims. Thus, the breadth and scope of the present invention should not be limited by any of the above-described exemplary embodiments, but should be defined only in accordance with the following claims and their equivalents.